RECEIVED
CENTRAL FAX CENTER
NOV 0 3 2004

PTO/SB/21 (09-04)
Approved for use through 07/31/2006. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

# TRANSMITTAL FORM

*(to be used for all correspondence after initial filing)*

Total Number of Pages in This Submission: 16

| | |
|---|---|
| Application Number | 09/539848 |
| Filing Date | 03/31/2000 |
| First Named Inventor | Donnelly |
| Art Unit | 2153 |
| Examiner Name | Flynn |
| Attorney Docket Number | AUS000116US1 |

## ENCLOSURES *(Check all that apply)*

- [x] Fee Transmittal Form
  - [ ] Fee Attached
- [ ] Amendment/Reply
  - [ ] After Final
  - [ ] Affidavits/declaration(s)
- [ ] Extension of Time Request
- [ ] Express Abandonment Request
- [ ] Information Disclosure Statement
- [ ] Certified Copy of Priority Document(s)
- [ ] Reply to Missing Parts/ Incomplete Application
  - [ ] Reply to Missing Parts under 37 CFR 1.52 or 1.53
- [ ] Drawing(s)
- [ ] Licensing-related Papers
- [ ] Petition
- [ ] Petition to Convert to a Provisional Application
- [ ] Power of Attorney, Revocation
- [ ] Change of Correspondence Address
- [ ] Terminal Disclaimer
- [ ] Request for Refund
- [ ] CD, Number of CD(s) _____
  - [ ] Landscape Table on CD
- [ ] After Allowance Communication to TC
- [ ] Appeal Communication to Board of Appeals and Interferences
- [x] Appeal Communication to TC (Appeal Notice, Brief, Reply Brief)
- [ ] Proprietary Information
- [ ] Status Letter
- [ ] Other Enclosure(s) (please identify below):

Remarks

## SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

| | | | |
|---|---|---|---|
| Firm Name | LALLY & LALLY, L.L.P. | | |
| Signature | | | |
| Printed name | Joseph P. Lally | | |
| Date | November 3, 2004 | Reg. No. | 38,947 |

## CERTIFICATE OF TRANSMISSION/MAILING

I hereby certify that this correspondence is being facsimile transmitted to the USPTO or deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on the date shown below:

| | | | |
|---|---|---|---|
| Signature | | | |
| Typed or printed name | Joseph P. Lally | Date | November 3, 2004 |

This collection of information is required by 37 CFR 1.5. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and1.14. This collection is estimated to 2 hours to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

*If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.*

PTO/SB/17 (10-04v2)
Approved for use through 07/31/2006. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

# FEE TRANSMITTAL for FY 2005

*Effective 10/01/2004. Patent fees are subject to annual revision.*

☐ Applicant claims small entity status. See 37 CFR 1.27

**TOTAL AMOUNT OF PAYMENT** ($) 340.00

| Complete If Known | |
|---|---|
| Application Number | 09/539848 |
| Filing Date | 03/31/2000 |
| First Named Inventor | Donnelly |
| Examiner Name | Flynn |
| Art Unit | 2153 |
| Attorney Docket No. | AUS000116US1 |

## METHOD OF PAYMENT (check all that apply)

☐ Check ☑ Credit card ☐ Money Order ☐ Other ☐ None

☐ Deposit Account:

Deposit Account Number
Deposit Account Name

The Director is authorized to: (check all that apply)
☐ Charge fee(s) indicated below ☐ Credit any overpayments
☐ Charge any additional fee(s) or any underpayment of fee(s)
☐ Charge fee(s) indicated below, except for the filing fee
to the above-identified deposit account.

## FEE CALCULATION

### 1. BASIC FILING FEE

| Large Entity Fee Code | Large Entity Fee ($) | Small Entity Fee Code | Small Entity Fee ($) | Fee Description | Fee Paid |
|---|---|---|---|---|---|
| 1001 | 790 | 2001 | 395 | Utility filing fee | |
| 1002 | 350 | 2002 | 175 | Design filing fee | |
| 1003 | 550 | 2003 | 275 | Plant filing fee | |
| 1004 | 790 | 2004 | 395 | Reissue filing fee | |
| 1005 | 160 | 2005 | 80 | Provisional filing fee | |

SUBTOTAL (1) ($)

### 2. EXTRA CLAIM FEES FOR UTILITY AND REISSUE

| | Extra Claims | | Fee from below | | Fee Paid |
|---|---|---|---|---|---|
| Total Claims | | -20** = | | X | = |
| Independent Claims | | - 3** = | | X | = |
| Multiple Dependent | | | | | = |

| Large Entity Fee Code | Large Entity Fee ($) | Small Entity Fee Code | Small Entity Fee ($) | Fee Description |
|---|---|---|---|---|
| 1202 | 18 | 2202 | 9 | Claims in excess of 20 |
| 1201 | 88 | 2201 | 44 | Independent claims in excess of 3 |
| 1203 | 300 | 2203 | 150 | Multiple dependent claim, if not paid |
| 1204 | 88 | 2204 | 44 | ** Reissue independent claims over original patent |
| 1205 | 18 | 2205 | 9 | ** Reissue claims in excess of 20 and over original patent |

SUBTOTAL (2) ($)

**or number previously paid, if greater; For Reissues, see above

## FEE CALCULATION (continued)

### 3. ADDITIONAL FEES

| Large Entity Fee Code | Large Entity Fee ($) | Small Entity Fee Code | Small Entity Fee ($) | Fee Description | Fee Paid |
|---|---|---|---|---|---|
| 1051 | 130 | 2051 | 65 | Surcharge - late filing fee or oath | |
| 1052 | 50 | 2052 | 25 | Surcharge - late provisional filing fee or cover sheet | |
| 1053 | 130 | 1053 | 130 | Non-English specification | |
| 1812 | 2,520 | 1812 | 2,520 | For filing a request for *ex parte* reexamination | |
| 1804 | 920* | 1804 | 920* | Requesting publication of SIR prior to Examiner action | |
| 1805 | 1,840* | 1805 | 1,840* | Requesting publication of SIR after Examiner action | |
| 1251 | 110 | 2251 | 55 | Extension for reply within first month | |
| 1252 | 430 | 2252 | 215 | Extension for reply within second month | |
| 1253 | 980 | 2253 | 490 | Extension for reply within third month | |
| 1254 | 1,530 | 2254 | 765 | Extension for reply within fourth month | |
| 1255 | 2,080 | 2255 | 1,040 | Extension for reply within fifth month | |
| 1401 | 340 | 2401 | 170 | Notice of Appeal | 340.00 |
| 1402 | 340 | 2402 | 170 | Filing a brief in support of an appeal | |
| 1403 | 300 | 2403 | 150 | Request for oral hearing | |
| 1451 | 1,510 | 1451 | 1,510 | Petition to institute a public use proceeding | |
| 1452 | 110 | 2452 | 55 | Petition to revive - unavoidable | |
| 1453 | 1,370 | 2453 | 685 | Petition to revive - unintentional | |
| 1501 | 1,370 | 2501 | 685 | Utility issue fee (or reissue) | |
| 1502 | 490 | 2502 | 245 | Design issue fee | |
| 1503 | 660 | 2503 | 330 | Plant issue fee | |
| 1460 | 130 | 1460 | 130 | Petitions to the Commissioner | |
| 1807 | 50 | 1807 | 50 | Processing fee under 37 CFR 1.17(q) | |
| 1806 | 180 | 1806 | 180 | Submission of Information Disclosure Stmt | |
| 8021 | 40 | 8021 | 40 | Recording each patent assignment per property (times number of properties) | |
| 1809 | 790 | 2809 | 395 | Filing a submission after final rejection (37 CFR 1.129(a)) | |
| 1810 | 790 | 2810 | 395 | For each additional invention to be examined (37 CFR 1.129(b)) | |
| 1801 | 790 | 2801 | 395 | Request for Continued Examination (RCE) | |
| 1802 | 900 | 1802 | 900 | Request for expedited examination of a design application | |

Other fee (specify) _____

*Reduced by Basic Filing Fee Paid SUBTOTAL (3) ($) 340.00

## SUBMITTED BY

(Complete (if applicable))

| | | | |
|---|---|---|---|
| Name (Print/Type) | Joseph P. Lally | Registration No. (Attorney/Agent) 38,947 | Telephone 512.428.9870 |
| Signature | | | Date November 3, 2004 |

**WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.**

This collection of information is required by 37 CFR 1.17 and 1.27. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

RECEIVED
CENTRAL FAX CENTER
NOV 0 3 2004

PATENT

IBM.5210R

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

| | | | |
|---|---|---|---|
| Applicant: | **Donnelly *et al.*** | Art Unit: | **2153** |
| Serial No.: | **09/539848** | Examiner: | **Flynn** |
| Filed: | **March 31, 2000** | Attorney Docket: | **AUS000116US1** |
| For: | **Remote Execution of Commands in a Multi-Host Network** | | |

I, the undersigned Joseph P. Lally, hereby certify that this correspondence is being facsimile transmitted to the USPTO or deposited with the US Postal Service with sufficient postage as first class mail in an envelope addressed to: MAIL STOP APPEAL BRIEF, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

November 3, 2004
Date

Signature

## APPEAL BRIEF

MAIL STOP APPEAL BRIEF
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

Dear Sir:

This paper is submitted pursuant to 37 CFR § § 41.31, 41.37 in furtherance of the Notice of Appeal filed on September 3, 2004 following a Final Office Action (the "Office Action") dated May 3, 2004, to appeal final rejections issued by the Examiner on claims in the above referenced patent application to the Board of Patent Appeals and Interferences ("Board").

~~11/05/2004 AWONDAF1 00000016 09539848~~
~~01 FC:1402 340.00 OP~~

11/05/2004 AWONDAF1 00000023 09539848
01 FC:1402 340.00 OP

TABLE OF CONTENTS

## I. REAL PARTY IN INTEREST

The above referenced application is wholly assigned to International Business Machines Corporation ("IBM"), A New York corporation having a principle place of business at Armonk, New York.

## II. RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences known to Appellant that will directly affect, be directly affected by, or have a bearing on the Board's decision in this appeal.

## III. STATUS OF CLAIMS

Claims 1-9, 12-16, and 19-23 are pending in this application. Claims 1-9, 12-16, and 19-23 stand rejected under the Office Action. More particularly:

1. Claim 1 stands rejected under 35 U.S.C. 103(a) as being unpatentable over Goldstein (U.S. Patent Number 6,374,287), hereinafter "Goldstein".

2. Claims 2-5 stand rejected under 35 U.S.C. 103(a) as being unpatentable over Goldstein in view of Lefebvre (U.S. Patent No. 6,249,294), hereinafter "Lefebvre".

3. Claims 6-9, 12, and 20-21 stand rejected under 35 U.S.C. 103(a) as being unpatentable over Lefebvre.

4. Claims 13-16, 19, and 22 stand rejected under 35 U.S.C. 103(a) as being unpatentable over Lefebvre in view of Goldstein.

5. Claim 23 stands rejected under 35 U.S.C. 103(a) as being unpatentable over Lefebvre in view of Giokas *et al.* (U.S. Patent No. 5,408,602), hereinafter "Giokas".

In this Appeal, Appellant is challenging the rejections of claims 1, 2, 4, 6, and 12. Appellant acknowledges that, for purposes of this appeal, claim 2 and 3 stand and fall together, claims 4 and 5 stand and fall together, claims 6-9, and 22 stand and fall together and claims 12-22 stand and fall together.

## IV. STATUS OF AMENDMENTS

No amendments have been filed subsequent to the final rejection.

## V. SUMMARY OF CLAIMED SUBJECT MATTER

Independent claim 1 is directed at multi-host data processing network (100) having a local host (102), a remote host (103), and a terminal (110). Terminal (110) includes a display (116), a keyboard (112), and a pointing device (114). A display server (101) on local host (102) is associated with a user of terminal (110). [Disclosure at Page 3, Line 17]. Display server (101) enables the user to execute GUI applications on local host (102) and remote host (103) from terminal (110) via a display server authorization mechanism [Disclosure at Page 4, line 25]. According to claim 1, network (100) enables the user to execute a text string entered at the terminal (110) as a shell command on the remote host (103) via the display server (101) [Disclosure at paragraph beginning on Page 6, line 29 and paragraph beginning on Page 7, line 26].

Independent claim 6 is directed at a distributed computer program product for enabling remote execution of a command string as shell command on a remote host (103) [Paragraph beginning on Page 7, line 26]. The claimed program product includes a display server (101) on local host (102). Display server (101) enables a user of a display terminal (110) to invoke local and remote GUI applications. [Paragraph beginning on page 3, line 11]. A client application (1076) on local host (102) is connected to display server (101). When client application (107) receives a command string, it pastes the command string to a clipboard [Paragraph beginning on Page 7, line 12]. A daemon process (210) on remote host (103) retrieves the command sting from the clipboard and executes the command string as a shell command on the remote host (103). [Page 6, lines 5-8; Paragraph beginning on Page 7, line 14].

Independent claim 12 is directed at a method of executing a shell command on a remote host. The claimed method includes creating (402) a first window with a first process and storing (404) an identifier (ID) associated with the first window (300) as a property (304) of display server (101). The display server property (304) is then monitored (406) for alterations. A command string is entered (408) using a client application (107). The client application (107) stores the command string in a clipboard (306) associated with first window (300) causing an alteration (410) in the display server property (304). Upon detecting the alteration in display server property (410), the command string is retrieved (412) from the clipboard (306) and executed (412) as a shell command on the remote host (103). [Paragraph beginning on Page 8, line 8].

## VI. GROUND OF REJECTION TO BE REVIEWED ON APPEAL

The grounds of rejection to be reviewed on appeal include:

1. Is claim 1 patentable over Goldstein under 35 U.S.C. 103(a)?
2. Is claim 2 patentable over Goldstein in view of Lefebvre under 35 U.S.C. 103(a)?
3. Is claim 4 patentable over Goldstein in view of Lefebvre under 35 U.S.C. 103(a)?

4. Are claims 6 and 12 patentable over Lefebvre under 35 U.S.C. 103(a)?

## VII. ARGUMENT

### Claim 1

The obviousness rejection of claim 1 is improper because the cited reference does not disclose all of the limitations of the claim. To establish *prima facie* obviousness of a claimed invention, all the claim limitations must be taught or suggested by the reference(s) cited. MPEP 2143.03 (citing *In re Royka*, 490 F.2d 981, 180 USPQ 580 (CCPA 1974)). In the present case, Goldstein does not disclose or suggest a network and windows server enabled to execute a text string entered at the terminal as a shell command on the remote host.

The Office Action states that Goldstein discloses a network "configured to enable the user to execute a text string entered at the terminal as a shell command on the remote host via the display server." The Office Action cites col. 4 lines 58-67 and col. 5 lines 1-5 as support for this assertion. The cited portion of Goldstein reads as follows:

> In step 201, the local window server 123 receives a request from the local client process 121 to perform an action. For example, the request may call for a user interface on the display device 103 to be updated in response to a button selection by a user. The window server examines the request and determines whether the request can be processed by program code residing on the local computer 101 or whether the request needs to be processed by a window server extension which resides on the remote computer 129 (step 203). If the request should be processed remotely (step 205), then the window server sends the request to the remote extension 145 residing on the remote computer 129 (step 207), where the remote extension processes the request in such a way that the output from the processing can itself be processed by the window server 123. (emphasis added).

Contrary to the assertion of the Office Action, the cited portion of Goldstein does not disclose a network that enables a user to execute a text string, entered at a terminal, as a shell command on a remote host. Goldstein discloses that a window server receives a request from a local client. The only example of such a request disclosed in Goldstein is a request to update a user interface on a display device in response to a *button selection by the user*.

A button selection is a classic example of a GUI action. Windows servers are specifically designed to process GUI actions across systems in a network. Thus, the portion

of Goldstein cited by the Office Action to support the rejection of claim 1 merely recites that a windows server may update a user interface in response to a GUI action performed by a user. The cited portion of Goldstein simply does not disclose or suggest a windows server implementation or extension that enables the user to execute locally entered text string as a command on a remote system. Goldstein, for example, does not use the word "text" even once and there is nothing in the Office Action to indicate how Goldstein's reference to a well-known use of a windows server discloses or suggests the text string limitation of claim 1. Accordingly, Appellant would respectfully submit that the Section 103(a) rejection of claim 1 based on Goldstein is improper and that, therefore, claim 1 and its dependent claims (claims 2-5) are allowable over the cited reference.

Claim 2

The Office Action rejected claim 2 as unpatentable over Goldstein in view of Lefebvre. The Office Action acknowledges that Goldstein does not disclose wherein the local host includes a client application and the remote host includes a daemon process and wherein the client application is enabled to receive the command from the user and the daemon process is configured to retrieve and execute the command. The Office Action attempts to support the Section 103(a) rejection of claim 2 by stating that Lefebvre discloses a system that includes "a client process operable to receive and broadcast OGL command buffers and daemon processes operable to access the shared memory structures for the purpose of executing the commands stored therein."

Assuming arguendo that the Office Action's summary of what Lefebvre disclose is accurate, that disclosure still does not teach or suggest the limitations of claim 2. Claim 2 plainly recites a client application that receives text string from the user and a daemon process that retrieves the text string and executes it as a shell command. According to the Office Action then, a client process operable to receive and broadcast OpenGL command buffers teaches a client process that receives text strings. But an OpenGL command buffer is not a text string and invoking a daemon process to execute commands stored in an OpenGL command buffer is not executing a command string as a shell command on a remote system. OpenGL is a graphics API and OpenGL commands perform graphics functionality. OpenGL graphics commands are

functionally related to windows servers since both are concerned with graphical displays in computer systems. A text string and text string command, to the contrary, are perhaps the paradigmatic opposites of a computer graphic and nothing describing a mechanism for executing OpenGL command buffers can be suggestive of a technique for executing text strings as shell commands on a remote system. Accordingly, Appellant submits that the Section 103(a) rejection of claim 2 is improper and that claim 2 and its dependent claims (claims 3-5) are, therefore, allowable over the cited references.

Claim 4

The Section 103(a) rejection of claim 4 is improper because the cited references do not disclose or suggest the all of the claimed limitations. Claim 4 recites that the daemon process opens a display server window and stores a window ID of the window as the display server property. Claim 4 further recites that the client application is configured to change the display server property to zero upon receiving the text string.

Supporting the obviousness rejection of claim 4, the Office Action states that col. 15 lines 37-42 of Lefebvre discloses wherein the daemon process is configured to open a display server window and to store a window id of the display window as the display server property. The cited portion of Lefebvre reads as follows:

> Similarly, in step 1306, master 310 issues an XMoveWindow(A1,-1020,600) command to slave 314, wherein A1 is the window ID on slave 314 that corresponds to the master window ID A, and -1020,600 are the offset versions of the coordinates passed from client 110 to master 310.

The office action further states that Lefebvre discloses wherein the client application is configured to alter the display server property to zero upon receiving the command from. Citing col. 15 lines 60-65. The cited portion of Lefebvre reads as follows:

> Otherwise, master 310 issues a walxEventGetExpose() request to the slave in step 1316. In response, the slave sends to the master all of the information necessary to define all of the expose events in its queue. (This operation also has the effect of clearing the expose events queue within that slave.)

The rejected claim recites that the display server property that is set to zero is the window ID of the window opened by the daemon process. In contrast, the cited portions of Lefebvre do not

disclose or suggest that a Window ID (A1 in the example) is set to zero (or otherwise altered at all) upon receiving a text string. Instead, Lefebvre discloses that expose events in a slave's queue are cleared when those expose events are delivered to the master. This portion of Lefebvre is not related to the portion of Lefebvre discussing window IDs. Thus, Lefebvre does teach or suggest all of the limitations of claim 4.

Claims 6 and 12

The obviousness rejections of claims 6 and 12 are improper because the cited references do not disclose all of the limitations of the claim. Specifically, Lefebvre does not teach or suggest a client application on a local host configured to receive a command string and a daemon process on a remote host configured to retrieve the command sting execute the command string as a shell command on the remote host.

In considering claims 6 and 12, the Office Action states that col. 19 lines 26-31 of Lefebvre disclose a client application on the local host that receives a command string and pastes the command string to a clipboard and that col. 19 lines 35-39 of Lefebvre disclose a daemon process on the remote host that retrieves the command string from the clipboard and executes the command string as a shell command. Column 19, lines 26-39 of Lefebvre read as follows:

> In order to make better use of network bandwidth when operating in the remote mode, OGL commands are buffered on the client side in a buffer such as OGL command buffer 1822. OGL command buffer 1822 is created and managed by client-side GLX library 1823. OGL commands are buffered on the server side in a buffer such as shared memory OGL command buffer 1824. GLX extension 1825 within X server process 1808 creates, manages and fills shared memory OGL command buffer 1824. OGL daemon 1810 has a shared memory connection to OGL command buffer 1824, as shown at 1826. It uses shared memory connection 1826 to read OGL command buffer 1824 so that it can execute the commands stored there.

Appellant is unable to determine how the cited portion of Lefebvre discloses the claimed limitations under discussion. The terminology command string as used in the present application is clearly and consistently confined to a text string. As such, the command string limitation of claim 6, especially when read in light of the claim limitation of executing the command string as a shell command, should be interpreted as a text-based command string.

In other words, because a shell command is a text string, the command string of claim 6 that is executed as a shell command must be a text string.

Under this reading of the claim, it is apparent that Lefebvre does not disclose a client application that receives a command string and pastes the command string to a clipboard, nor does Lefebvre disclose a daemon process that retrieves command strings from a clipboard and executes them as a shell command. Like Goldstein, Lefebvre does not describe or suggest the use of a windows server in conjunction with text-based commands.

Furthermore, the Office Action acknowledges that, even under a broad interpretation of the claim language "command string," Lefebvre does not disclose executing the command string as a shell command. Supporting the rejection, nevertheless, the Office Action states that executing shell commands on a remote host are well known Telnet features. According, concludes the Office Action, it would have been obvious to incorporate Telnet protocol into the Lefebvre system because it would expand the compatibility and usability of the system for executing remote commands.

There is, however, no motivation to modify Lefebvre to incorporate a Telnet style protocol because the windows server system because Lefebvre is not directed at user oriented commands. The relevant portions of Lefebvre are concerned with a method and system for execution graphic commands on the graphics hardware of a remote host in a way that achieves desirable performance. Thus, the relevant portions of the Lefebvre, including the portions cited by the Examiner, describe the system by which Lefebvre enables OpenGL commands issued by a local application to be executed in graphics hardware on a remote host. It would not be obvious to modify a method and system for executing OpenGL commands in a multi-host network environment to execute user entered, text-based command strings. Lefebvre is not remotely interested in executing shell commands on remote systems, but is instead concerned with performing graphics command buffers in an efficient manner.

Moreover, as Appellant argued in response to a prior non-final office action, a Telnet style protocol, even if incorporated, would not achieve the results obtained with the claimed invention because the windows server system and the Telnet system would require different authorization mechanisms. The present invention achieves a method and system by

which the conventional uses of windows server systems are extended to incorporate text string execution.

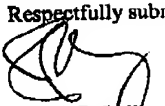
In applications where a display server system is already present, it is undesirable to require an additional application such as Telnet to enable shell command execution. Moreover, in host systems that implement access authorization, display server access privileges are independent of the Telnet access privileges. Thus, in a Telnet implementation, a user would be required to gain access (password, etc.) to the display server to invoke distributed GUI functionality and further required to acquire access to the remote server desktop to execute remote shell commands using another mechanism, such as a Telnet logon sequence. Therefore, one would not look to a Telnet application as a means of modifying the windows server system of Lefebvre to incorporate the text string execution capability of the claimed invention.

Accordingly, Appellant submits that the Section 103(a) rejection of claims 6 and 12 are improper and that claims 6 and 12 and their dependent claims are allowable over the cited references.

## *CONCLUSION*

In view of the foregoing, Applicant would submit that the pending claims are allowable over the cited references and would respectfully request the Board to remand this application to the Examiner for reconsideration consistent with an order that the Examiner pass this case to issuance unless a proper rejection to the claims can be made.

Respectfully submitted,



Joseph P. Lally
Reg. No. 38,947
ATTORNEY FOR APPLICANT(S)

P.O. Box 684749
Austin, Texas 78768-4749
512.428.9870
512.428.9871 (FAX)

JPL/mnun

## VIII. CLAIMS APPENDIX

### TEXT OF CLAIMS PRESENTED ON APPEAL:

1 (previously presented). A multi-host data processing network including a local host and a remote host, comprising:

a terminal including a display, a keyboard, and a pointing device;

a display server on the local host associated with a user of the terminal, wherein the display server enables the user to execute GUI applications on the local and remote hosts from the terminal via a display server authorization mechanism; and

wherein the network is configured to enable the user to execute a text string entered at the terminal as a shell command on the remote host via the display server.

2 (previously presented). The network of claim 1, wherein the local host includes a client application and the remote host includes a daemon process, wherein the client application is enabled to receive the text string from the user and the daemon process is configured to retrieve the text string and execute the text string as the shell command.

3 (previously presented). The network of claim 2, wherein the daemon process is configured to monitor changes to a property of the display server, and further wherein the client application is configured to alter the display server property upon receiving the text string.

4 (previously presented). The network of claim 3, wherein the daemon process is configured to open a display server window and to store a window id of the display server window as the display server property and wherein the client application is configured to change the display server property to zero upon receiving the text string.

5 (previously presented). The network of claim 4, wherein the client application is enabled to transfer the text string to a clipboard associated with the display server window and wherein the daemon process is enabled to retrieve the text string from the clipboard upon detecting a change to the display server property.

6 (previously presented). A distributed network windowing system computer program product enabling remote execution in a data processing network including a local host and a remote host, the computer product comprising:

a display server on the local host, wherein the display server enables a user of a display terminal connected to the network to invoke local and remote GUI applications;

a client application on the local host connected to the display server and configured to receive a command string and, upon receiving the command string, to paste the command string to a clipboard; and

a daemon process on the remote host configured to retrieve the command sting from the clipboard and further configured to execute the command string as a shell command on the remote host.

7 (previously presented). The computer program product of claim 6, wherein the daemon process is configured, upon initiation, to open a display window in the display server and to store an id associated with the display window in a property of the display server.

8 (previously presented). The computer program product of claim 7, wherein the client application is configured, upon receiving the command string, to read the display server property, and to re-set the display server property to zero.

9 (original). The computer program product of claim 8, wherein the daemon process is configured to retrieve the command string from the clipboard responsive to detecting an alteration to the display server property.

10-11 (canceled).

12 (previously presented). A method of executing a shell command on a remote host, comprising:

creating a first window with a first process;

storing an id associated with the first window as a property of a display server;

monitoring for alterations in the display server property with the first process;

entering a command string via a client application, wherein, upon receiving the command string, the client application is configured to store the command string in a clipboard associated with the first window and to alter the display server property; and

upon detecting the alteration in the display server property, retrieving the command string from the clipboard and executing the command string as a shell command on the remote host.

13 (previously presented). The method of claim 12, wherein the first process resides on the remote host of a multi-host data processing system, the client resides on a local host of the system, and the command string is entered on a terminal connected to the local host.

14 (original). The method of claim 12, wherein the command comprises a command shutting down the remote host.

15 (original). The method of claim 12, wherein the command comprises a command invoking an application residing on the remote host.

16 (original). The method of claim 12, wherein the command string is entered by a user of a terminal controlled by the display server.

17-18 (canceled).

19 (original). The method of claim 12, further comprising, prior to creating the first window, logging into a local host of a multi-host computer system, wherein upon logging in, a windowing system initiates the display server and creates an authorization file associated with a user and by which applications connect to the display server.

20 (original). The method of claim 12, wherein the display server enables execution of local and remote GUI applications from a terminal served by the display server.

21 (original). The method of claim 12, wherein the altering of the display server property by the client application comprises the client application writing the display server property to zero.

22 (original). The method of claim 12, wherein the display server comprises an X Server of an X Window System.

23 (previously presented). The computer program product of claim 6, wherein the client application and the daemon host have access privilege to the display server via a common access code associated with the user and wherein the user is enabled to execute the shell command on the remote host based solely on the access privilege to the display server.